

LAPORAN PRAKTIKUM PEMROGRAMAN WEB 1
MODUL 6
JQUERY
(Dosen Pengampu: Yati Nurhayati S.Kom, M.Kom.)



Nama : **Adi awaludin fatah**
NIM : **20230810138**
Kelas : **TINFC-2023-04**

FAKULTAS ILMU
KOMPUTER UNIVERSITAS
KUNINGAN 2024

PEMBAHASAN

Pre Test:

Fungsi dari jQuery: jQuery adalah sebuah library JavaScript yang memudahkan manipulasi HTML/DOM, penanganan event, animasi, dan interaksi dengan server melalui AJAX. Beberapa keuntungan menggunakan jQuery antara lain:

1. **Memudahkan Seleksi Elemen:** Dengan sintaks yang lebih ringkas, jQuery memungkinkan kamu untuk memilih elemen HTML dengan mudah menggunakan selektor CSS.
2. **Penanganan Event:** Mempermudah pengaturan event seperti klik, hover, input, dll.
3. **Manipulasi DOM:** Mengubah konten, atribut, gaya (CSS), atau struktur halaman dengan cepat.
4. **Efek Animasi:** Menambahkan efek animasi seperti show, hide, fade, dan slide dengan kode minimal.
5. **AJAX:** Memudahkan komunikasi dengan server tanpa harus me- refresh halaman.

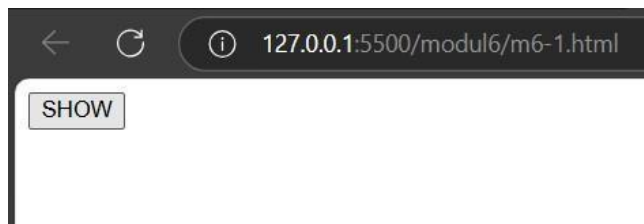
Buat script menggunakan jquery untuk menampilkan dan menyembunyikan gambar

```

modul6 > <> pretest.html > ...
1  <!DOCTYPE html>
2  <html lang="id">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Menampilkan dan Menyembunyikan Gambar dengan jQuery</title>
7      <!-- Menyertakan library jQuery -->
8      <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
9      <style>
10         body {
11             font-family: Arial, sans-serif;
12             text-align: center;
13             margin-top: 50px;
14         }
15         button {
16             margin-top: 20px;
17             padding: 10px 20px;
18             font-size: 16px;
19             background-color: #007bff;
20             color: white;
21             border: none;
22             border-radius: 5px;
23             cursor: pointer;
24         }
25         button:hover {
26             background-color: #0056b3;
27         }
28         img {
29             margin-top: 20px;
30             max-width: 300px;
31             display: none; /* Awalnya gambar disembunyikan */
32         }
33     </style>
34 </head>
35 <body>
36     <button id="toggleImage">Tampilkan Gambar</button>
37     <br>
38     <!-- Gambar yang akan ditampilkan atau disembunyikan -->
39     
40
41     <script>
42         // jQuery Code
43         $(document).ready(function () {
44             // Mengubah teks tombol dan menampilkan/menyembunyikan gambar
45             $("#toggleImage").click(function () {
46                 $("#gambar").toggle(); // Toggle menampilkan/menyembunyikan gambar
47                 // Mengubah teks tombol sesuai dengan status gambar
48                 if ($("#gambar").is(":visible")) {
49                     $("#toggleImage").text("Sembunyikan Gambar");
50                 } else {
51                     $("#toggleImage").text("Tampilkan Gambar");
52                 }
53             });
54         });
55     </script>
56 </body>
57 </html>

```

Output:



Analisis:

Source code ini merupakan implementasi sederhana dari manipulasi elemen HTML menggunakan jQuery untuk menyembunyikan dan menampilkan elemen. Ketika halaman dimuat, tombol dengan kelas tampil secara otomatis disembunyikan menggunakan metode `.hide()`. Ada dua tombol interaktif, yaitu tombol HIDE dengan kelas sembunyi dan tombol SHOW dengan kelas tampil. Ketika tombol HIDE diklik, elemen `<div>` yang berisi teks "UNIVERSITAS KUNINGAN" akan disembunyikan secara perlahan menggunakan efek animasi `hide("slow")`, tombol SHOW akan ditampilkan menggunakan metode

`.show()`, dan tombol HIDE akan disembunyikan. Sebaliknya, ketika tombol SHOW diklik, elemen teks tersebut akan kembali muncul secara perlahan menggunakan efek `show("slow")`, tombol HIDE akan kembali terlihat, dan tombol SHOW akan disembunyikan.

Kode ini memanfaatkan pustaka jQuery, yang memungkinkan pengembang memanipulasi elemen DOM dengan sintaks yang lebih sederhana dibandingkan JavaScript murni. Fungsi utama diletakkan dalam event

`$(document).ready()` untuk memastikan semua elemen halaman telah dimuat sebelum fungsi-fungsi dijalankan. Dengan penggunaan metode seperti `.hide()`,

`.show()`, dan efek animasi transisi seperti "slow", interaksi menjadi lebih dinamis. Namun, desain halaman ini cukup minimalis karena tidak menggunakan CSS tambahan untuk memperindah tampilan tombol atau

elemen teks. Untuk pengembangan lebih lanjut, gaya visual dan elemen interaktif lainnya dapat ditambahkan agar lebih menarik.

3. JQuery

```

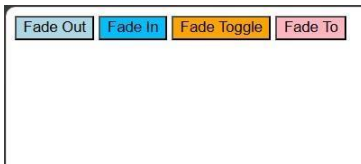
modul6 > m6-2.html > html > head > script > ready() callback
1  <!DOCTYPE html>
2  <html lang="id">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6      <title>jQuery</title>
7      <!-- file js-->
8      <script src="jquery-3.7.1.min.js"></script>
9
10     <script>
11       $(document).ready(function () {
12         $(".fadeout").click(function () {
13           $(".img").fadeOut();
14         });
15
16         $(".fadein").click(function () {
17           $(".img").fadeIn();
18         });
19
20         $(".fadetoggle").click(function () {
21           $(".img").fadeToggle();
22           $(".img").css("background-color", "orange");
23         });
24
25         $(".fadeto").click(function () {
26           $(".img").fadeTo("slow", 0.5);
27           $(".img").css("background-color", "lightpink");
28         });
29       });
30     </script>
31   </head>
32   <body>
33     <button class="fadeout" style="background-color: #add8e6">Fade Out</button>
34     <button class="fadein" style="background-color: #00bfff">Fade In</button>
35     <button class="fadetoggle" style="background-color: #ffa500">
36       Fade Toggle
37     </button>
38
39     <button class="fadeto" style="background-color: #ffb6c1">Fade To</button>
40
41     <br>
42     
43   </body>
44 </html>

```

Output:



Fade out:



Fade in:



Fade Toggle:



Pada Fade Toggle seharusnya background logonya itu berwarna orange, karena ukuran foto yang kebesaran jadi ga terlalu kelihatan Fado

To:



Analisis

Struktur HTML

1. Elemen Kepala (<head>):

- Menggunakan atribut dasar seperti meta untuk pengaturan encoding (UTF-8) dan responsivitas (viewport).
- Mengimpor pustaka jQuery versi 3.7.1 untuk mendukung manipulasi DOM.

2. Elemen Tombol:

- Terdapat empat tombol dengan kelas berbeda:
 - **Fade Out** (.fadeout): Mengurangi visibilitas gambar hingga sepenuhnya hilang.
 - **Fade In** (.fadein): Mengembalikan visibilitas gambar yang sebelumnya tersembunyi.
 - **Fade Toggle** (.fadetoggle): Mengubah visibilitas gambar (berpindah antara tampak atau tersembunyi).

- **Fade To** (.fadeto): Mengatur transparansi gambar ke tingkat tertentu (50% dalam kasus ini).

3. Elemen Gambar:

- Menggunakan tag untuk menampilkan gambar dengan sumber file gambar/logo_prodi_TI.png.

Fungsi-Fungsi jQuery

1. **Fade Out** (\$(".fadeout").click(function () { ... })):

- Ketika tombol "Fade Out" diklik, gambar akan menghilang secara bertahap menggunakan metode fadeOut().

2. **Fade In** (\$(".fadein").click(function () { ... })):

- Ketika tombol "Fade In" diklik, gambar yang sebelumnya disembunyikan akan muncul kembali secara bertahap menggunakan metode fadeIn().

3. **Fade Toggle** (\$(".fadetoggle").click(function () { ... })):

- Ketika tombol "Fade Toggle" diklik, gambar akan berpindah antara terlihat dan tersembunyi menggunakan metode fadeToggle().
- Selain itu, properti CSS gambar diubah dengan menambahkan warna latar belakang oranye (background-color: "orange").

4. **Fade To** (\$(".fadeto").click(function () { ... })):

- Ketika tombol "Fade To" diklik, transparansi gambar akan diatur ke 50% secara bertahap menggunakan metode fadeTo("slow", 0.5).
- Selain itu, properti CSS gambar diubah menjadi warna latar belakang merah muda (background-color: "lightpink").

Interaktivitas

- **Efek Visual:** Semua tombol memberikan efek animasi pada elemen gambar. Transisi yang halus membuat interaksi lebih menarik.
- **Dinamika Gaya (CSS):** Beberapa tombol mengubah warna latar belakang gambar menggunakan metode css(). Hal ini memberikan dimensi visual tambahan selain efek *fade*.

4. JQuery


```

modul6 > <> m6-3.html > html > body
 1  <!DOCTYPE html>
 2  <html lang="id">
 3  <head>
 4      <meta charset="UTF-8">
 5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
 6      <title>jQuery</title>
 7      <script src="jquery-3.7.1.min.js"></script>
 8      <style>
 9          .tab{
10              display: none;
11          }
12          .tab-button{
13              cursor: pointer;
14              padding: 10px;
15              background-color: #f1f1f1;
16              border : 1px solid #ccc;
17              display: inline-block;
18          }
19          .active{
20              background-color: #ccc;
21          }
22      </style>
23  </head>
24  <body>
25      <h1>Tugas Web Developer</h1>
26      <div>
27          <div class="tab-button active" data-tab="tab1">Tab 1</div>
28          <div class="tab-button" data-tab="tab2">Tab 2</div>
29          <div class="tab-button" data-tab="tab3">Tab 3</div>
30      </div>
31      <div id="tab1" class="tab" style="display: block;">
32          <h2>Front-End Developer</h2>
33          <p>Mengoptimalkan Fungsi Aplikasi berdasarkan User Interface (UI)
34              dan use Experience</p>
35      </div>
36      <div id="tab2" class="tab">
37          <h2>Back-End Development</h2>
38          <p>Gejala di sisi server dengan mengelola data
39              menjadi informasi dari database</p>
40      </div>
41      <div id="tab3" class="tab">
42          <h2>Full-stack Development</h2>
43          <p>Memiliki kemampuan di kedua sisi, yaitu Front-End
44              dan back-end, sehingga dapat menangani semua
45              aspek pengembang web </p>
46      </div>
47
48      <script>
49          $(document).ready(function(){
50              $(".tab-button").click(function(){
51                  $(".tab").hide();
52
53                  $(".tab-button").removeClass("active");
54
55                  tabToShow = $(this).data("tab");
56                  $("#" + tabToShow).show();
57                  $(this).addClass("active");
58              })
59          })
60      </script>
61  </body>
62  </html>

```

```

55      tabToShow = $(this).data("tab");
56      $("#" + tabToShow).show();
57      $(this).addClass("active");
58  })
59  })
60  </script>
61 </body>
62 </html>

```

Output:

Tab 1



Tab 2



Tab 3



Analisis:

Source code ini adalah implementasi sederhana dari sistem navigasi tab menggunakan jQuery. Halaman ini menampilkan tiga tab dengan konten berbeda: "Front-End Developer," "Back-End Development," dan "Full- stack Development." Ketika sebuah tab diklik, konten terkait ditampilkan, sementara konten dari tab lain disembunyikan. Sistem

navigasi ini menggunakan elemen HTML seperti `<div>` untuk tombol tab (`.tab-button`) dan konten tab (`.tab`). Secara default, hanya konten tab pertama (`#tab1`) yang ditampilkan saat halaman dimuat, dengan gaya CSS untuk menyembunyikan semua tab lainnya (`display: none`).

Interaktivitas halaman diatur menggunakan jQuery. Ketika pengguna mengklik sebuah tombol tab, fungsi JavaScript pertama-tama

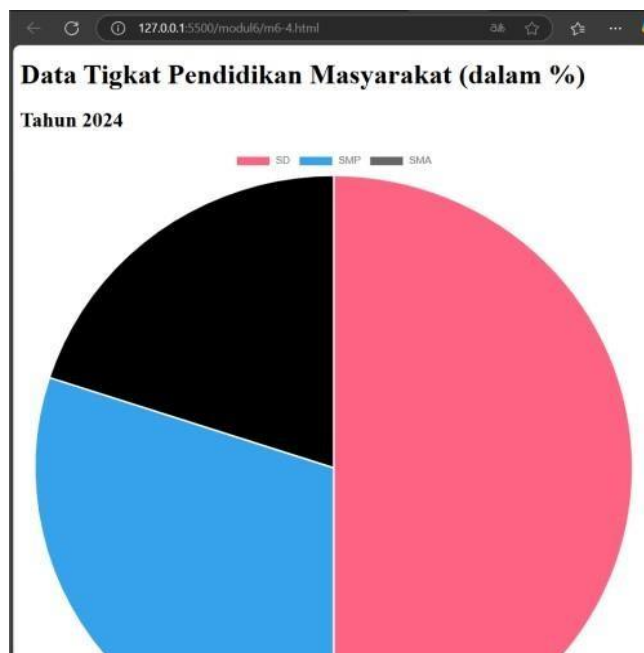
menyembunyikan semua konten tab menggunakan `.hide()` dan menghapus kelas `active` dari semua tombol tab. Kemudian, fungsi menampilkan konten tab yang sesuai berdasarkan atribut `data-tab` tombol yang diklik dan menambahkan kelas `active` pada tombol tersebut untuk memberikan efek visual berupa perubahan warna latar belakang.

Desain antarmuka menggunakan CSS dasar, memberikan tombol tab tampilan dengan padding, warna latar belakang netral, dan efek *hover* sederhana. Kelas `active` digunakan untuk menonjolkan tab yang sedang dipilih. Implementasi ini sederhana, efektif, dan cocok untuk menampilkan informasi yang dikelompokkan dalam beberapa kategori tanpa perlu memuat ulang halaman. Sistem ini dapat dikembangkan lebih lanjut dengan menambahkan animasi transisi atau responsivitas untuk meningkatkan pengalaman pengguna.

5. Pie Chart dan Bar Chart

```
modul6 > <> m6-4.html > html
1  <!DOCTYPE html>
2  <html lang="id">
3  <head>
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <title>Pie Chart Bar Chart</title>
7    <script src="jquery-3.7.1.min.js"></script>
8    <script src="chart.js"></script>
9  </head>
10 <body>
11   <h1>Data Tingkat Pendidikan Masyarakat (dalam %)</h1>
12   <h2>Tahun 2024</h2>
13   <canvas id="myPieChart"></canvas>
14
15   <script>
16     $(document).ready(function () {
17       //Data untuk Pie Chart
18       pieData = {
19         labels: ["SD", "SMP", "SMA"],
20         datasets: [
21           {
22             data: [50, 30, 20],
23             backgroundColor: ["#FF6384", "#36A2EB", "#FFCE56"],
24           },
25         ],
26       };
27
28       //Membuat Pie Chart
29       ctxPie = document.getElementById("myPieChart").getContext("2d");
30       myPieChart = new Chart(ctxPie, {
31         type: "pie",
32         data: pieData,
33       });
34     });
35   </script>
36 </body>
37 </html>
```

Output:



Analisis:

Kode HTML dan JavaScript ini menampilkan sebuah *pie chart* untuk merepresentasikan data tingkat pendidikan masyarakat dalam bentuk persentase pada tahun 2024. Grafik ini dibuat menggunakan pustaka Chart.js, yang diimpor melalui elemen `<script>`. Saat halaman dimuat, jQuery menginisialisasi data yang akan digunakan untuk grafik, terdiri dari tiga kategori pendidikan: SD (50%), SMP (30%), dan SMA (20%). Warna-warna yang berbeda diterapkan pada setiap kategori untuk membedakan segmen pada *pie chart*: merah muda untuk SD, biru untuk SMP, dan kuning untuk SMA.

Grafik ini dirender dalam elemen `<canvas>` dengan ID `myPieChart`. Chart.js memanfaatkan data yang telah ditentukan dalam variabel `pieData` untuk menghasilkan visualisasi *pie chart*. Struktur data meliputi labels untuk kategori pendidikan dan datasets untuk nilai persentase serta warna setiap kategori. Metode `getContext("2d")` digunakan untuk menggambar grafik dua dimensi pada elemen `<canvas>`.

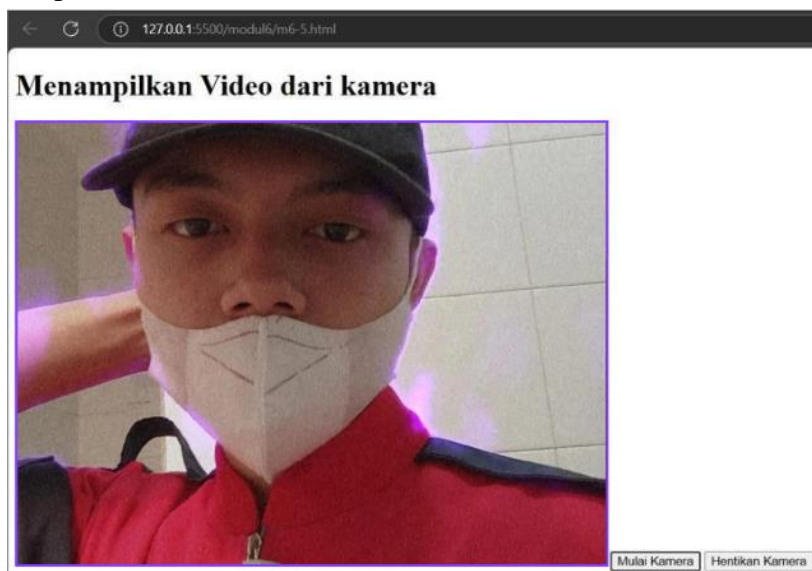
Ketika halaman diakses, pengguna akan melihat judul "Data Tingkat Pendidikan Masyarakat (dalam %)" dan "Tahun 2024," diikuti dengan sebuah *pie chart*. Grafik ini membagi lingkaran menjadi tiga segmen sesuai dengan persentase data: bagian terbesar (50%) diwakili oleh SD, segmen sedang (30%) oleh SMP, dan segmen terkecil (20%) oleh SMA. Warna segmen membuat grafik lebih mudah dibaca dan

memberikan gambaran langsung tentang distribusi tingkat pendidikan masyarakat.

6. Akses Kamera dengan JQuery

```
modul6 > m6-5.html > html > body > script > ready() callback > click() callback > catch() c
1 <!DOCTYPE html>
2 <html lang="id">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Akses Kamera dengan JQuery</title>
7   <!-- file js -->
8   <script src="jquery-3.7.1.min.js"></script>
9 </head>
10 <body>
11   <h1>Menampilkan Video dari kamera</h1>
12   <video id="video" width="640" height="480" autoplay></video>
13   <button id="start">Mulai Kamera</button>
14   <button id="stop">Hentikan Kamera</button>
15   <script>
16     $(document).ready(function(){
17       const video = document.getElementById("video");
18       let mediaStream;
19
20       $("#start").click(function(){
21         navigator.mediaDevices
22           .getUserMedia({video: true })
23           .then(function (stream){
24             mediaStream = stream;
25             video.srcObject = stream;
26           })
27           .catch(function (err){
28             console.error("Gagal mengakses kamera: " + err);
29           });
30       });
31       $("#stop").click(function(){
32         if (mediaStream){
33           mediaStream.getTracks().forEach((track) => track.stop());
34           video.srcObject = null;
35         }
36       });
37     });
38   </script>
39 </body>
40 </html>
```

Output:



Analisis:

Kode ini merupakan implementasi sederhana untuk mengakses kamera perangkat menggunakan JavaScript, dipadukan dengan jQuery untuk mempermudah interaktivitas. Halaman ini berisi elemen `<video>` yang berfungsi untuk menampilkan video langsung dari kamera, serta dua tombol untuk memulai dan menghentikan akses kamera. Fungsi utama diatur menggunakan API `getUserMedia`, bagian dari `navigator.mediaDevices`, yang memungkinkan aplikasi web mengakses perangkat keras seperti kamera dengan izin dari pengguna.

Ketika tombol "Mulai Kamera" diklik, fungsi `getUserMedia` meminta akses ke kamera perangkat. Jika akses diberikan, aliran video ditampilkan pada elemen `<video>` melalui properti `srcObject`. Apabila permintaan akses ditolak atau terjadi kesalahan, pesan kesalahan akan ditampilkan di konsol. Sebaliknya, ketika tombol "Hentikan Kamera" diklik, aliran video dihentikan dengan memanggil metode `stop()` pada setiap *track* aliran media, dan elemen `<video>` dikosongkan dengan mengatur `srcObject` menjadi `null`.

Fungsi dan Output

- **Fungsi Mulai Kamera:** Tombol "Mulai Kamera" menampilkan video langsung dari kamera perangkat di elemen `<video>`.
- **Fungsi Hentikan Kamera:** Tombol "Hentikan Kamera" menghentikan aliran video dan melepaskan sumber daya kamera.
- **Kesalahan yang Ditangani:** Jika kamera tidak tersedia atau akses ditolak, pesan kesalahan akan muncul di konsol, membantu pengembang dalam debugging.

7. Manipulasi DOM dengan JQuery


```

modul6 > <m6-6.html> <html> <head> <script>
1 <!DOCTYPE html>
2 <html lang="id">
3   <head>
4     <meta charset="UTF-8" />
5     <meta name="viewport" content="width=device-width, initial-scale=1.0 /" />
6     <title>Manipulasi DOM dengan JQuery</title>
7     <script src="jquery-3.7.1.min.js"></script>
8   </head>
9   <body>
10    <h1>Manipulasi DOM dengan JQuery</h1>
11
12    <button id="tambah">Tambah Data</button>
13    <button id="ubah">Ubah Data Pertama</button>
14    <button id="hapus">Hapus Data Terakhir</button>
15
16    <div id="output">
17      <div class="item">Data 1</div>
18      <div class="item">Data 2</div>
19      <div class="item">Data 3</div>
20    </div>
21
22    <script>
23      $(document).ready(function () {
24        // Menambahkan Elemen
25        $("#tambah").on("click", function () {
26          $("#output").append('<div class="item">Data Baru</div>');
27        });
28
29        // Mengubah Elemen Pertama
30        $("#ubah").on("click", function () {
31          $("#output .item:first").text("Data Pertama Diubah");
32        });
33
34        // Menghapus Elemen Pertama
35        $("#hapus").on("click", function () {
36          $("#output .item:last").remove();
37        });
38      });
39    </script>
40  </body>
41 </html>

```

Output:



Tambah data



Ubah Data Pertama



Hapus Data Terakhir



Analisis:

Kode ini adalah implementasi manipulasi DOM (Document Object Model) menggunakan jQuery untuk memodifikasi elemen pada halaman web secara dinamis. Terdapat tiga tombol dengan fungsi berbeda untuk menambah, mengubah, dan menghapus elemen di dalam div dengan ID output. Elemen yang dimanipulasi adalah div-div dengan kelas item, yang awalnya terdiri dari tiga elemen: "Data 1," "Data 2," dan "Data 3."

Penjelasan Fungsi

1. Tambah Data:

- Tombol "Tambah Data" dengan ID tambah menggunakan metode `.append()` untuk menambahkan elemen baru (`<div class="item">Data Baru</div>`) di akhir elemen `#output`.
- Setiap kali tombol diklik, data baru akan terus ditambahkan ke daftar elemen.

2. Ubah Data Pertama:

- Tombol "Ubah Data Pertama" dengan ID ubah menggunakan metode `.text()` untuk mengganti teks dari elemen pertama (`:first`) dalam `#output` dengan "Data Pertama Diubah."
- Hanya elemen pertama yang diubah, tidak memengaruhi elemen lainnya.

3. Hapus Data Terakhir:

- Tombol "Hapus Data Terakhir" dengan ID hapus menggunakan metode `.remove()` untuk menghapus elemen terakhir (`:last`) dalam `#output`.
- Jika tidak ada elemen yang tersisa, fungsi ini tidak menimbulkan kesalahan tetapi tidak melakukan apa pun.

Interaktivitas dan Output

- **Sebelum Interaksi:** Elemen #output berisi tiga div (Data 1, Data 2, dan Data 3).
- **Setelah Interaksi:**
 - Mengklik tombol "Tambah Data" menambahkan elemen baru dengan teks "Data Baru" di akhir.
 - Mengklik tombol "Ubah Data Pertama" mengubah teks elemen pertama menjadi "Data Pertama Diubah."
 - Mengklik tombol "Hapus Data Terakhir" menghapus elemen terakhir dalam daftar.
- Pengguna dapat melakukan kombinasi klik untuk melihat hasil manipulasi secara dinamis.

8. Penerapan event Handling dengan JQuery

```

modul6 > <> m6-7.html > html > head > style > .box
1  <!DOCTYPE html>
2  <html lang="id">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Penerapan Event Handling dengan JQuery</title>
7      <script src="jquery-3.7.1.min.js"></script>
8      <style>
9          body{
10             font-family: Arial, sans-serif;
11             margin: 20px;
12         }
13         .box{
14             width: 100px;
15             height: 100px;
16             background-color: lightblue;
17             margin: 10px;
18             display: inline-block;
19             text-align: center;
20             line-height: 100px;
21             font-weight: bold;
22             cursor: pointer;
23             transition: background-color 0.3s;
24         }
25     </style>
26 </head>
27 <body>
28     <h1>Penerapan Event Handling dengan JQuery</h1>
29
30     <div class="box" id="box1">Box 1</div>
31     <div class="box" id="box2">Box 2</div>
32     <div class="box" id="box3">Box 3</div>
33
34     <script>
35         $(document).ready(function(){
36             // Event Click
37             $(".box").on("click", function(){

```

```

38         $(this).toggleClass("active");
39         alert('Anda mengklik ${$(this).text()}');
40     });
41
42     // Event mouseenter
43     $(".box").on("mouseenter", function(){
44         $(this).css("background-color", "lightgreen");
45     });
46
47     // Event mouseleave
48     $(".box").on("mouseleave", function(){
49         $(this).css("background-color", "lightblue");
50     });
51
52     });
53
54
55
56 </script>
57 </body>
58 </html>

```

Output:



Analisis:

Kode ini mengimplementasikan beberapa event handling menggunakan jQuery untuk elemen-elemen interaktif dalam bentuk kotak yang dapat diklik dan berinteraksi dengan gerakan mouse. Terdapat tiga elemen

<div> yang masing-masing diberi kelas box dan memiliki teks yang menunjukkan nama kotak (Box 1, Box 2, dan Box 3). Setiap kotak ini diberi gaya CSS yang membuatnya tampil sebagai kotak berwarna biru muda dengan teks di tengah dan efek transisi pada perubahan warna latar belakang.

Tiga jenis event ditangani dalam kode ini. Pertama, event click diatur dengan menggunakan .on("click"), yang menambahkan atau menghapus kelas active pada kotak yang diklik. Ketika kotak diklik, akan muncul pesan pop-up yang menunjukkan nama kotak yang diklik.

Kedua, event

mouseenter diatur dengan .on("mouseenter"), yang mengubah warna

latar belakang kotak menjadi hijau terang ketika kursor mouse berada di atasnya. Ketiga, event `mouseleave` menggunakan `.on("mouseleave")` untuk mengembalikan warna latar belakang kotak ke biru muda ketika kursor mouse meninggalkan kotak.

Dengan menggunakan jQuery untuk menangani event-event ini, kode ini memungkinkan elemen-elemen kotak untuk bereaksi terhadap interaksi pengguna dengan cara yang dinamis dan responsif, menciptakan pengalaman pengguna yang lebih interaktif dan menarik.

```
1 <!DOCTYPE html>
2 <html lang="id">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Form Pemesanan</title>
7   <link rel="stylesheet" href="Latihan.css">
8   <script src="jquery-3.7.1.min.js"></script>
9 </head>
10 <body>
11   <form>
12     <h1>Form Pemesanan</h1>
13     <div id="container">
14       <div id="label">
15         <label for="input">Id Pemesanan</label>
16         <label for="input">Nama Pemesan</label>
17         <label for="input">Nomor Identitas</label>
18         <label for="">Jenis Kelamin</label>
19         <label for="input">Tipe Kamar</label>
20         <label for="input">Harga</label>
21         <label for="input">Tanggal Pemesan</label>
22         <label for="input">Durasi Menginap</label>
23         <label for="input">Termasuk Breackfast</label>
24         <label for="input">Total Bayar</label>
25       </div>
26       <div id="inputan">
27         <input type="number" name="Id Pemesanan" min="0" required/>
28         <input type="text" name="Nama" required/>
29         <input type="number" id="Nomor Identitas" min="0" required/>
30         <div id="JK">
```

Latihan /

Tugas Souce

code: Html

```

31         <input type="radio" name="jeniskelamin"
32         value="laki-laki" required>laki-laki
33         <input type="radio" name="jeniskelamin" value="Perempuan" required>Perempuan
34     </div>
35
36
37     <select name="tipe_kamar" id="Tipe_Kamar">
38         <option value="">--Pilih--</option>
39         <option value="Standar">Standar</option>
40         <option value="Deluxe">Deluxe</option>
41         <option value="Family">Family</option>
42     </select>
43     <input type="number" min="0" placeholder="Rp." id="Harga" readonly>
44     <input type="date" id="Tanggal Pesan">
45     <input type="number" id="durasi" min="0" placeholder="Hari" required/>
46     <div id="makansiang">
47         <input type="checkbox" style="height: 10px; accent-color:blue;" id="Breakfast" value="ya">Ya
48     </div>
49     <input type="text" id="Totalbayar" placeholder="Rp." readonly>
50 </div>
51 </div>
52 <div class="button">
53     <input type="button" name="save" value="SIMPAN" id="simpan" onclick="Resume()">
54     <input type="button" name="cancel" value="BATAL" id="batal">
55 </div>
56 <div id="resume">
57 </div>
58 </form>
59
60 <script>
61     $(document).ready(function () {
62         // Hitung Total Bayar
63         $('#simpan').on('click', function () {
64             const tipeKamar = $('#Tipe_Kamar').val();
65             const hargaKamar = parseInt($('#Tipe_Kamar option:selected').data('harga')) || 0; // Pastikan data harga ada di option
66             const durasiMenginap = parseInt($('#durasi').val()) || 0;
67             const includeBreakfast = $('#Breakfast').is(':checked');
68
69             if (!tipeKamar || durasiMenginap <= 0) {
70                 alert("Pilih tipe kamar dan durasi menginap dengan benar!");
71                 return;
72             }
73
74             // Hitung total
75             let totalBayar = hargaKamar * durasiMenginap;
76             if (durasiMenginap > 3) {
77                 totalBayar *= 0.9; // Diskon 10%
78             }
79             if (includeBreakfast) {
80                 totalBayar += 100000;
81             }
82
83             // Tampilkan total bayar
84             $('#Totalbayar').val('Rp ' + totalBayar.toLocaleString());
85         });
86
87         // Simpan dan tampilkan resume
88         $('#simpan').on('click', function () {
89             const idPemesanan = $('#input[name="Id Pemesanan"]').val();
90             const namaPemesan = $('#input[name="Nama"]').val();
91             const jenisKelamin = $('#input[name="jeniskelamin"]:checked').val();
92             const nomorIdentitas = $('#Nomor\\ Identitas').val();
93             const tanggalPesan = $('#Tanggal\\ Pesan').val();
94             const tipeKamar = $('#Tipe_Kamar').val();
95             const durasiMenginap = $('#durasi').val();
96             const includeBreakfast = $('#Breakfast').is(':checked');
97             const totalBayar = $('#Totalbayar').val();
98
99             alert(`
100             Resume Pemesanan:
101             ID Pemesanan: ${idPemesanan}
102             Nama Pemesan: ${namaPemesan}
103             Nomor Identitas: ${nomorIdentitas}
104             Jenis Kelamin: ${jenisKelamin}
105             Tanggal Pesan: ${tanggalPesan}
106             Tipe Kamar: ${tipeKamar}
107             Durasi Menginap: ${durasiMenginap} hari
108             Termasuk Breakfast: ${includeBreakfast ? "Ya" : "Tidak"}
109             Total Bayar: ${totalBayar}
110             `);
111         });
112     });
113 </script>
114 </body>
115 </html>

```

Css

```

1  body{
2      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
3      background-color:gainsboro;
4  }
5  #container{
6      display:flex ;
7  }
8  #label{
9      margin-left: 20px;
10     display:flex;
11     flex-direction: column;
12     gap: 20px;
13     width: 250px;
14 }
15 #inputan{
16     display: flex;
17     padding-left: 30px;
18     flex-direction: column;
19     gap: 20px;
20     width: 500px;
21 }
22 .JK{
23     display: flex;
24 }
25 input{
26     border: none;
27     border-radius: 5px;
28     height: 20px;
29 }
30 select{
31     border: none;
32     border-radius: 5px;
33     height: 20px;
34 }
35 .button{
36     display: flex;
37     gap: 10px;
38     padding-left: 300px;
39     padding-top: 30px;
40 }
41 #simpan{
42     background-color: blue;
43     color: white;
44     height: 30px;
45     width: 70px;
46     border: none;
47     border-radius: 10px;
48 }
49 #batal{
50     background-color:orangered;
51     color: white;
52     height: 30px;
53     width: 70px;
54     border-radius: 10px;
55 }
56 #simpan:hover{
57     height: 50px;
58     width: 90px;
59 }
60 #batal:hover{
61     height: 50px;
62     width: 90px;
63 }

```

Output:

Form Pemesanan

| | |
|---------------------|--|
| Id Pemesanan | <input type="text" value="001"/> |
| Nama Pemesan | <input type="text" value="Eka"/> |
| Nomor Identitas | <input type="text" value="1234"/> |
| Jenis Kelamin | <input checked="" type="radio"/> laki-laki <input type="radio"/> Perempuan |
| Tipe Kamar | <input type="text" value="Deluxe"/> |
| Harga | <input type="text" value="Rp."/> |
| Tanggal Pemesan | <input type="text" value="12/02/2024"/> |
| Durasi Menginap | <input type="text" value="7"/> |
| Termasuk Breackfast | <input checked="" type="checkbox"/> Ya |
| Total Bayar | <input type="text" value="Rp 100,000"/> |

127.0.0.1:5500 says

Resume Pemesanan:
ID Pemesanan: 001
Nama Pemesan: Eka
Nomor Identitas: 1234
Jenis Kelamin: laki-laki
Tanggal Pesan: 2024-12-02
Tipe Kamar: Deluxe
Durasi Menginap: 7 hari
Termasuk Breakfast: Ya
Total Bayar: Rp 100,000

Analisis

HTML (Struktur Halaman)

HTML digunakan untuk membangun form pemesanan kamar. Berikut adalah elemen utama dan fungsinya:

a. Form Input

- **Id Pemesanan** dan **Nomor Identitas**: Input berupa angka untuk mempermudah validasi dan menghindari kesalahan format.
- **Nama Pemesan**: Input teks yang memungkinkan pengguna mengisi nama mereka.
- **Jenis Kelamin**: Radio button untuk memilih antara **laki-laki** atau

perempuan. Hanya satu pilihan yang bisa dipilih.

- **Tipe Kamar:** Dropdown (select) dengan tiga pilihan kamar: **Standar, Deluxe, dan Family.** Namun, atribut **harga tiap kamar** belum ditambahkan.
- **Harga, Tanggal Pemesan, Total Bayar:** Input **readonly** digunakan untuk menampilkan data hasil kalkulasi (seperti total bayar) atau data otomatis.
- **Durasi Menginap dan Termasuk Breakfast:** Input angka untuk menentukan jumlah hari menginap dan checkbox untuk memilih apakah termasuk sarapan.

b. Tombol:

- **SIMPAN:** Untuk menyimpan dan menghitung data.
- **BATAL:** Tidak memiliki fungsi spesifik dalam kode ini, tetapi biasanya digunakan untuk mereset form atau membatalkan pengisian

CSS (Tampilan dan Tata Letak)

CSS memberikan gaya pada form agar lebih rapi dan menarik:

a. Layout:

- **Flexbox:** Elemen-elemen seperti label dan input diatur menggunakan **flexbox** agar lebih rapi. **flex-direction: column** digunakan untuk menyusun elemen secara vertikal.
- **Lebar Elemen:** Label diberi lebar tetap (250px), sedangkan input diberi lebar lebih besar (500px) untuk menyesuaikan teks yang diisi.

b. Estetika:

- Warna tombol:
 - Tombol **SIMPAN** diberi warna biru untuk memberi kesan aksi positif.
 - Tombol **BATAL** diberi warna oranye untuk menarik perhatian agar tidak salah klik.
- Hover efek:
 - Ukuran tombol berubah saat kursor diarahkan ke tombol, memberikan respons visual kepada pengguna.
- Font:
 - Menggunakan font modern seperti Segoe UI dan Tahoma untuk meningkatkan keterbacaan.

Javascript (Fungsi Interaktif)

Javascript digunakan untuk menghitung dan menampilkan total pembayaran, serta menampilkan resume pemesanan.

a. Logika Perhitungan Total Bayar:

1. Mengambil Data:

- **Tipe Kamar:** Nilai diambil dari dropdown.
- **Durasi Menginap:** Diambil dari input angka.
- **Include Breakfast:** Cek apakah checkbox sarapan dicentang.

2. Hitung Total:

- **Total dihitung** berdasarkan harga kamar, durasi, dan tambahan biaya sarapan.
- **Jika durasi menginap > 3 hari**, total mendapatkan diskon 10%.

3. Tampilkan Total Bayar: Hasil total ditampilkan dalam input readonly dengan format Rp xxx.xxx.

b. Menampilkan Resume Pemesanan:

- Resume informasi seperti ID Pemesanan, Nama Pemesan, dan Total Bayar ditampilkan menggunakan **alert** dalam format yang rapi.